

IN THE CLAIMS

1 1. [currently amended] A process comprising:
2 ~~for~~ automatically translating a formal language specification written in a formal
3 language defining a full and complete computer program to be automatically written by a
4 computer into a full and complete source code computer program that can be compiled
5 into a complete, executable program which can execute by itself on a computer and
6 needs no additional third party source code or source code from existing components or
7 code libraries to be compiled with it to make said complete executable program and which
8 implements the requirements of said formal language specification, said formal language
9 specification defining at least classes of objects having attributes, services and
10 relationships with other classes, ~~said specification written in a formal language,~~
11 comprising the following steps:
12 using a computer, automatically write computer code that will request user
13 name and password, receive any responses and authenticate the user;
14 using a computer, automatically write computer code that will determine
15 this user's privilege level and query said formal language specification and
16 determine all object attributes this user has privilege to see and query and all
17 services this user can invoke;
18 using a computer, automatically write computer code which queries said
19 formal language specification for all services of all classes that any authorized
20 user may invoke and identifies an object server which will implement each said
21 service;
22 using a computer, automatically write code that will retrieve service
23 arguments for all services from one or more of a user, an ~~or from another~~ object
24 server, and ~~or from~~ another process, as appropriate;

25 using a computer, automatically write code that controls a computer to
26 display displays menus options, icons or creates any other means by which and
27 entity a user or another process can invoke a service, and which receives input
28 to invoke a particular service and responds by sending a message to the
29 appropriate object server to invoke the service, said message including the
30 necessary arguments for the service to execute;

31 using a computer, automatically write code that implements an object
32 server for every service, each of which first checks to verify that state
33 transitions are valid and make sense for the current state of objects of which the
34 object service will be altering the state ~~of~~;

35 using a computer, automatically write code for every object server that
36 verifies preconditions are satisfied before making state transitions of any objects
37 the states of which are acted upon by the object server;

38 using a computer, automatically write code to make all valuation
39 calculations required by said formal language specification of each object server;

40 using a computer, automatically write code to verify that integrity
41 constraints specified in said formal language specification on the values of
42 attributes of objects have been satisfied after execution of a service and take
43 action if said integrity constraints are not satisfied; and

44 using a computer, automatically write code for every object server to test
45 trigger relationships specified in said formal language specification after
46 execution of a service and carry out appropriate action if a trigger event has
47 occurred.

1 2. [currently amended] An apparatus for automatically translating a formal

2 specification written in a formal language which has predefined rules of grammar, said
3 formal specification defining a computer program to be automatically written by a
4 computer, said translating acting to convert said formal specification into a computer
5 program that implements the requirements of said formal specification, said formal
6 specification defining at least classes of objects having attributes, services and
7 relationships with other classes, ~~said specification written in a formal language,~~
8 comprising:

9 a computer programmed with an operating system and one or more other
10 programs to cooperate with said operating system to control said computer to perform
11 the following functions:

12 automatically write computer code that will request user name and password,
13 receive any responses and authenticate the user;

14 automatically write computer code that will determine a this user's privilege level
15 and query said formal ~~language~~ specification and determine all object attributes said this
16 user has privilege to see and query and all services said this user can invoke;

17 automatically write computer code which queries said formal specification for all
18 services of all classes that any authorized user may invoke and identifies an object
19 server which will implement said service;

20 automatically write code that will retrieve service arguments for all services from
21 one or more of a user, or from another object server, and or from another process, as
22 appropriate;

23 automatically write code that displays one or more user interface tools menus
24 ~~options, icons or creates any other means by which a user or another process can~~
25 ~~invoke which can be used to invoke~~ a service, and which receives input to invoke a
26 particular service and which responds by sending a message to the appropriate object

server to invoke said ~~the~~ service, said message including the necessary arguments for said ~~the~~ service to execute;

automatically write code that implements an object server for every service, each of which first checks to verify that state transitions are valid and make sense for the current state of objects the object service will be altering the state of;

automatically write code for every object server that verifies preconditions are satisfied before making state transitions of any objects the states of which are acted upon by said ~~the~~ object server;

automatically write code to make all valuation calculations required by said specification of each object server;

automatically write code to verify that integrity constraints specified in said formal specification on the values of attributes of objects have been satisfied after execution of a service and take action if said integrity constraints are not satisfied; and

automatically write code for every object server to test trigger relationships specified in said formal specification after execution of a service and carry out appropriate action if a trigger event has occurred.

3. [currently amended] A computer-readable medium containing instructions for controlling a computer to automatically translate a formal specification defining a computer program to be automatically written by a computer into a computer program that implements the requirements of said formal specification, said formal specification defining at least classes of objects having attributes, services and relationships with other classes, said formal specification written in a formal language having predefined rules of grammar, by:

automatically writing computer code that will request user name and password,

9 receive any responses and authenticate the user;

10 automatically writing computer code that will determine a ~~this~~ user's privilege level
11 and query said formal ~~language~~ specification and determine all object attributes said ~~this~~
12 user has privilege to see and all services said ~~this~~ user can invoke;

13 automatically writing computer code which queries said formal specification for all
14 services of all classes that any authorized user may invoke and identifies an object
15 server which will implement said service;

16 automatically writing computer code that will retrieve service arguments for all
17 services from one or more of a user, an ~~or from another~~ object server, and ~~or from~~
18 another process, as appropriate;

19 automatically write code that displays menus options, icons or creates any other
20 means by which a user or another process can invoke a service, and which receives
21 input to invoke a particular service and responds by sending a message to the
22 appropriate object server to invoke the service, said message including the necessary
23 arguments for the service to execute;

24 automatically writing code that implements an object server for every service,
25 each of which first checks to verify that state transitions are valid and make sense for
26 the current state of objects the object service will be altering the state of;

27 automatically write code for every object server that verifies preconditions are
28 satisfied before making state transitions of any objects the states of which are acted
29 upon by the object server;

30 automatically write code to make all valuation calculations required by said formal
31 specification of each object server;

32 automatically write code to verify that integrity constraints specified in said formal
33 specification on the values of attributes of objects have been satisfied after execution of

34 a service and take action if said integrity constraints are not satisfied; and
35 automatically write code for every object server to test trigger relationships
36 specified in said formal specification after execution of a service and carry out
37 appropriate action if a trigger event has occurred.
38